

PTO/SB/21 (08-03)

Approved for use through 08/30/2003. OMB 0651-0031

U.S. Patent and Trademark Office; U.S. DEPARTMENT OF COMMERCE

Under the Paperwork Reduction Act of 1995, no persons are required to respond to a collection of information unless it displays a valid OMB control number.

**TRANSMITTAL  
FORM**

(to be used for all correspondence after initial filing)

Total Number of Pages in This Submission

52

Application Number

10/692,493

Filing Date

10/24/03

First Named Inventor

Yuji Suga

Art Unit

2124

Examiner Name

Attorney Docket Number

CFA00034US

**ENCLOSURES (Check all that apply)**

- ☐ Fee Transmittal Form
- ☐ Fee Attached
- ☐ Amendment/Reply
- ☐ After Final
- ☐ Affidavits/declaration(s)
- ☐ Extension of Time Request
- ☐ Express Abandonment Request
- ☐ Information Disclosure Statement
- ☒ Certified Copy of Priority Document(s)
- ☐ Response to Missing Parts/  
Incomplete Application
- ☐ Response to Missing Parts  
under 37 CFR 1.52 or 1.53

- ☐ Drawing(s)
- ☐ Licensing-related Papers
- ☐ Petition
- ☐ Petition to Convert to a  
Provisional Application
- ☐ Power of Attorney, Revocation  
Change of Correspondence Address
- ☐ Terminal Disclaimer
- ☐ Request for Refund
- ☐ CD, Number of CD(s) \_\_\_\_\_

- ☐ After Allowance communication  
to Technology Center (TC)
- ☐ Appeal Communication to Board  
of Appeals and Interferences
- ☐ Appeal Communication to TC  
(Appeal Notice, Brief, Reply Brief)
- ☐ Proprietary Information
- ☐ Status Letter
- ☐ Other Enclosure(s) (please  
Identify below):

Remarks

**SIGNATURE OF APPLICANT, ATTORNEY, OR AGENT**

Firm  
or  
Individual name

Canon U.S.A., Inc. IP Department  
Fidel Nwamu

Signature

Date

**CERTIFICATE OF TRANSMISSION/MAILING**

I hereby certify that this correspondence is being facsimile transmitted to the USPTO or deposited with the United States Postal Service with sufficient postage as first class mail in an envelope addressed to: Commissioner for Patents, P.O. Box 1450, Alexandria, VA 22313-1450 on the date shown below.

Typed or printed name

Fidel Nwamu

Signature

Date

2/11/04

This collection of information is required by 37 CFR 1.5. The information is required to obtain or retain a benefit by the public which is to file (and by the USPTO to process) an application. Confidentiality is governed by 35 U.S.C. 122 and 37 CFR 1.14. This collection is estimated to 12 minutes to complete, including gathering, preparing, and submitting the completed application form to the USPTO. Time will vary depending upon the individual case. Any comments on the amount of time you require to complete this form and/or suggestions for reducing this burden, should be sent to the Chief Information Officer, U.S. Patent and Trademark Office, U.S. Department of Commerce, P.O. Box 1450, Alexandria, VA 22313-1450. DO NOT SEND FEES OR COMPLETED FORMS TO THIS ADDRESS. SEND TO: Commissioner for Patents, P.O. Box 1450, Alexandria, VA 22313-1450.

If you need assistance in completing the form, call 1-800-PTO-9199 and select option 2.

日 本 国 特 許 庁  
JAPAN PATENT OFFICE

別紙添付の書類に記載されている事項は下記の出願書類に記載されている事項と同一であることを証明する。

This is to certify that the annexed is a true copy of the following application as filed with this Office.

出 願 年 月 日                      2 0 0 2 年 1 0 月 2 9 日  
Date of Application:

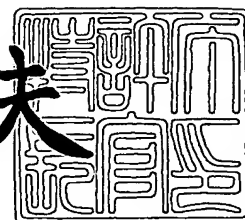
出 願 番 号                      特 願 2 0 0 2 - 3 1 4 5 9 5  
Application Number:  
[ST. 10/C]:                      [ J P 2 0 0 2 - 3 1 4 5 9 5 ]

出      願      人                      キヤノン株式会社  
Applicant(s):

2 0 0 3 年 1 1 月 1 8 日

特許庁長官  
Commissioner,  
Japan Patent Office

今 井 康 夫



【書類名】 特許願

【整理番号】 4823011

【提出日】 平成14年10月29日

【あて先】 特許庁長官 太田 信一郎 殿

【国際特許分類】 G06F 17/00

【発明の名称】 べき乗演算装置

【請求項の数】 1

【発明者】

    【住所又は居所】 東京都大田区下丸子3丁目30番2号キャノン株式会社  
                                内

    【氏名】 須賀 祐治

【特許出願人】

    【識別番号】 000001007

    【住所又は居所】 東京都大田区下丸子3丁目30番2号

    【氏名又は名称】 キャノン株式会社

    【代表者】 御手洗 富士夫

    【電話番号】 03-3758-2111

【代理人】

    【識別番号】 100090538

    【住所又は居所】 東京都大田区下丸子3丁目30番2号キャノン株式会社  
                                内

    【弁理士】

    【氏名又は名称】 西山 恵三

    【電話番号】 03-3758-2111

**【選任した代理人】****【識別番号】** 100096965**【住所又は居所】** 東京都大田区下丸子 3 丁目 3 0 番 2 号キャノン株式会  
社内**【弁理士】****【氏名又は名称】** 内尾 裕一**【電話番号】** 03-3758-2111**【手数料の表示】****【予納台帳番号】** 011224**【納付金額】** 21,000円**【提出物件の目録】****【物件名】** 明細書 1**【物件名】** 図面 1**【物件名】** 要約書 1**【包括委任状番号】** 9908388**【プルーフの要否】** 要

【書類名】 明細書

【発明の名称】 べき乗演算装置

【特許請求の範囲】

【請求項 1】 2つの整数  $x$  および  $e$  から  $x^e$  を計算するべき乗演算装置において、

2つの整数  $x$  および  $e$  を入力する入力手段と、

$L$  個の候補指数  $\{l_i\}$  ( $0 \leq i \leq L-1$ ) を格納する候補指数格納手段と

、  
前記候補指数格納手段に格納されている候補指数  $\{l_i\}$  のそれぞれに対し、  
入力された整数  $x$  から  $x^{l_i}$  を事前に計算しておく事前計算手段と、

事前計算された数値  $x^{l_i}$  を格納する事前計算数値格納手段と、

入力された整数  $e$  を前記候補指数  $\{l_i\}$  のいずれかと一致するように複数の  
数値  $\{f_i\}$  ( $0 \leq i \leq F-1$ ) に分割する分割手段と、

計算結果  $c$  を格納する計算結果格納手段と、

分割された数値  $\{f_i\}$  ( $0 \leq i \leq F-1$ ) について、事前計算された数値  
 $x^{l_i}$  を用いて計算結果  $c$  を逐次更新する逐次処理手段と、

全ての前記数値  $\{f_i\}$  について更新後の計算結果  $c$  を  $x^e$  として出力する  
出力手段と

を有することを特徴とするべき乗演算装置。

【発明の詳細な説明】

【0001】

【発明の属する技術分野】

本発明は、べき乗剰余演算などにおけるべき乗演算を行なう、べき乗演算装置に関するものである。

【0002】

【従来の技術】

べき乗剰余演算  $x^e \pmod{N}$  は、RSA暗号／署名をはじめ ElGamal暗号、DSA署名、Diffie-Hellman鍵共有方式などで利用されている演算である。ファイルに対する署名や復号などの用途だけではなくSS

Lなどの通信路のセキュア化においても利用されており、通信要求に対してインタラクティブに演算を行う必要があり、処理効率は暗号処理計算時間に大きく影響する。

#### 【0003】

べき乗剰余演算は a) 平方剰余演算  $x^2 \pmod{N}$  と、b) 乗算剰余演算  $xu \pmod{N}$  とにより構成することができる。与えられた  $e$  に対し  $x^e \pmod{N}$  を計算するために必要とする a) および b) の乗算演算の回数を削減することにより、全体の処理時間を高速化する手法がいくつか提案されている。

#### 【0004】

加算鎖 (addition chain) とは、 $a_1 = 1$  からはじまり最終的に  $a_n = e$  に至る整数の列であり、 $a_i$  はそれ以前に現れた数の和 (つまり  $a_i = a_j + a_k$  ( $j, k < i$ )) を満たす。例えば  $e = 55$  のとき  $\{1, 2, 3, 6, 12, 13, 26, 27, 54, 55\}$  はその例である。これは  $x \rightarrow x^2 \rightarrow x^3 \rightarrow x^6 \rightarrow x^{12} \rightarrow x^{13} \rightarrow x^{26} \rightarrow x^{27} \rightarrow x^{54} \rightarrow x^{55}$  のように a) および b) の演算を行うことで、 $x^{55}$  が計算できることを表している。これは  $\{1, 2, 3, 4, \dots, 52, 53, 54, 55\}$  のように b) の演算だけを用いるよりも計算量を少なくすることができることを意味している。このように、与えられた指数  $e$  (上の例では 55) に対し、より短い加算鎖を見つけるアルゴリズムが有益である。

#### 【0005】

##### [Binary Method]

Binary Method は上記のモチベーションに基づいたアルゴリズムの一つであり、D. E. Knuth, The Art of Computer Programming: Seminumerical Algorithms, volume 2, Reading, MA: Addison-Wesley, Second edition (1981) で紹介されている。

#### 【0006】

Binary Method は、次のような処理を行うアルゴリズムである。

与えられた指数  $e$  ( $k$  ビット長) を  $\sum_{i=0, \dots, k-1} 2^i * e\_i$  ( $e\_i$  は 0 または 1) のように 2 進表現しておく。入力を  $x$ ,  $e$ ,  $N$  とし、出力  $C$  は  $x^e \pmod{N}$  となるアルゴリズムは以下のとおりである。

#### 【0007】

```

1) if e_(k-1) = 1 then C := x else C := 1
2) for i = k-2 downto 0
    2-1) C := C * C (mod N)
    2-2) if e_i = 1 then C := C * x (mod N)
3) return C

```

上記において、2) の `for` は変数  $i$  が  $k-2$  から 0 まで 1 ずつ削減ながら 2-1) 及び 2-2) をループ処理することを表している。図 2 は  $e = 55$  のとき `Binary Method` を使って上記のように  $x^{55} \pmod{N}$  が計算される過程を表したものである。このときの加算鎖は  $\{1, 2, 3, 6, 12, 13, 26, 27, 54, 55\}$  である。

#### 【0008】

##### [m-ary Method]

`m-ary Method` は `Binary Method` を拡張させたもので、2 ビット以上の処理を一度に行う方式である。入力を  $x$ ,  $e$ ,  $N$  とし、出力  $C$  は  $x^e \pmod{N}$  となるアルゴリズムは以下の通りである。ただし、与えられた指数  $e$  は  $k$  ビット長であり、 $e$  は  $s$  個の  $r (= \log_2 m)$  ビット列  $F\_0, \dots, F\_s$  (ただし  $s$  は  $k/r$  を超えない整数) に分解されているとする。

#### 【0009】

```

0) w = 2, ..., m-1 について x^w (mod N) を事前に計算しておく
1) C := x^{F_(s-1)} (mod N) とおく。
2) for i = s-2 downto 0
    2-1) C := C^m (mod N)
    2-2) if F_i != 0 then C := C * x^{F_i}
(mod N)

```

3) return C

【0010】

m-ary Method は、 $m=4$  のとき Quaternary Method と呼ばれる。 $e=55$  のときの Quaternary Method の処理を図3に示す。 $E$  を2進数表示すると  $(110111)_2$  であるが、これを  $r=2$  ビットずつに分解すると、 $(\underline{11} \ \underline{01} \ \underline{11})_2$  であり、図3に示されている通りに処理される。このときの加算鎖は  $\{1, 2, 3, 6, 12, 13, 26, 52, 55\}$  であり、Binary Method に比べて加算鎖の長さが1つ短いため、 $x^{55}$  を計算するための剰余演算が少なくて済むことを意味している。

【0011】

さらに m-ary Method の拡張として、Slide Window Techniques など多くの改良案が提案されている。Slide Window Techniques では、固定長でなく、アルゴリズム中の2) の処理で1度に扱うビット長を可変長にすることで、アルゴリズム中の0) の処理にあたる事前計算処理を削減し、計算量の削減と事前計算結果を格納しておく領域(テーブルと呼ぶ)の削減を実現している。

【0012】

【発明が解決しようとする課題】

上述した従来技術において、Binary Method は事前計算を行う必要がなく、事前計算結果を格納するテーブルを持つ必要がないというメリットを持つ。しかし  $e$  を2進数表示したとき、1が立っているビットの個数が多い場合、計算処理を増大させてしまう欠点がある。また、Quaternary Method や Slide Window Techniques では、計算量を少なくする利点があるが、テーブル参照が必要となり事前計算処理の計算量がかさむデメリットがある。

【0013】

本発明は、 $e$  を2進数表示した際のビット列内にある特徴を持つビット列が登場することを想定し、それらのビット列に関してのみ事前計算を行うことで、事



前計算処理量とテーブルサイズを削減し、従来方式に比べて少ない演算回数で計算することができるべき乗計算方式を提供する。またテーブルサイズをおさえるメリットもあわせ持ち、テーブル参照のためのメモリ領域を削減することができる。

#### 【0014】

##### 【課題を解決するための手段】

そこで、本発明の目的は、べき乗演算における事前計算処理量とテーブルサイズとを低減させ、少ない演算回数で計算することができるべき乗演算装置を提供することにある。

#### 【0015】

上記目的を達成するために、本発明では、2つの整数  $x$  および  $e$  から  $x^e$  を計算するべき乗演算装置に、2つの整数  $x$  および  $e$  を入力する入力手段と、 $L$  個の候補指数  $\{l\_i\}$  ( $0 \leq i \leq L-1$ ) を格納する候補指数格納手段と、前記候補指数格納手段に格納されている候補指数  $\{l\_i\}$  のそれぞれに対し、入力された整数  $x$  から  $x^{l\_i}$  を事前に計算しておく事前計算手段と、事前計算された数値  $x^{l\_i}$  を格納する事前計算数値格納手段と、入力された整数  $e$  を前記候補指数  $\{l\_i\}$  のいずれかと一致するように複数の数値  $\{f\_i\}$  ( $0 \leq i \leq F-1$ ) に分割する分割手段と、計算結果  $c$  を格納する計算結果格納手段と、分割された数値  $\{f\_i\}$  ( $0 \leq i \leq F-1$ ) について、事前計算された数値  $x^{l\_i}$  を用いて計算結果  $c$  を逐次更新する逐次処理手段と、全ての前記数値  $\{f\_i\}$  について更新後の計算結果  $c$  を  $x^e$  として出力する出力手段とを備える。

#### 【0016】

##### 【発明の実施の形態】

以下、図面を参照して、本発明の実施形態を説明する。

#### 【0017】

##### (第1の実施の形態)

本発明は、例えば、図1に示すような情報処理装置(ホストコンピュータ) 100に適用される。本実施形態の情報処理装置 100は、パーソナルコンピュー

タ等のコンピュータを含み、視覚復号型秘密分散法により、機密画像情報から複数の分散画像情報を生成して分配する機能を実現する。

#### 【0018】

すなわち、情報処理装置100は、図1に示すように、公衆回線等のモデム118、表示部としてのモニタ102、CPU103、ROM104、RAM105、HD106、ネットワークのネットワーク接続部107、CD108、FD109、DVD110、プリンタ115のインターフェース（I/F）117、及び操作部としてのマウス112やキーボード113等のインターフェース（I/F）111が、バス116を介して互いに通信可能に接続された構成としている。

#### 【0019】

マウス112及びキーボード113は、ユーザが情報処理装置100に対する各種指示等を入力するための操作部である。この入力情報（操作情報）は、インターフェース111を介して情報処理装置100内に取り込まれる。

#### 【0020】

情報処理装置100での各種情報（文字情報や画像情報等）は、プリンタ115により印刷出力できるようになされている。

#### 【0021】

モニタ102は、ユーザへの各種指示情報や、文字情報或いは画像情報等の各種情報の表示を行う。

#### 【0022】

CPU103は、情報処理装置100全体の動作制御を司る。すなわち、CPU103は、HD（ハードディスク）106等から処理プログラム（ソフトウェアプログラム）を読み出して実行することで、情報処理装置100全体を制御する。特に、本実施形態では、CPU103は、機密画像情報から複数の分散画像情報を生成して分配するための処理プログラムを、HD106等から読み出して実行することで、後述する処理（以下、「情報分散処理」とも言う）を実施する。

#### 【0023】

ROM104は、必要な画像処理のための処理プログラム、及び情報分散処理のための処理プログラム等の各種処理プログラムや、各種データ等を記憶する。

【0024】

RAM105は、CPU103での各種処理のために一時的に処理プログラムや処理対象の情報を格納するための作業用エリア等として使用される。

【0025】

HD106は、大容量記憶装置の一例としての構成要素であり、文字情報や画像情報、或いは各種処理の実行時にRAM105等へ転送される情報分散処理等のための処理プログラム等を保存する。

【0026】

CD（CDドライブ）108は、外部記憶媒体の一例としてのCD（CD-R）に記憶されたデータを読み込み、また、当該CDへデータを書き出す機能を有する。

【0027】

FD（フロッピー（R）ディスクドライブ）109は、CD108と同様に、外部記憶媒体の一例としてのFDに記憶されたデータを読み込み、また、当該FDへデータを書き出す機能を有する。

【0028】

DVD（デジタルビデオディスクドライブ）110は、CD108やFD109と同様に、外部記憶媒体の一例としてのDVDに記憶されたデータを読み込み、また、当該DVDへデータを書き出す機能を有する。

【0029】

尚、CD、FD、DVD等の外部記憶媒体に対して、例えば、編集用のプログラム或いはプリンタドライバが記憶されている場合、これらをHD106へインストールし、必要に応じてRAM105へ転送するように構成してもよい。

【0030】

インターフェース（I/F）111は、マウス112やキーボード113によるユーザからの入力を受け付けるためのものである。

【0031】

モデム 118 は、通信モデムであり、インターフェース (I/F) 119 を介して、例えば公衆回線等を通じて外部のネットワークに接続される。

#### 【0032】

ネットワーク接続部 107 は、インターフェース (I/F) 114 を介して外部のネットワークに接続される。

#### 【0033】

図 4 は、図 1 の情報処理装置 100 において、特徴とする機能（情報分散処理の機能）に着目して図示したものである。情報処理装置 100 は、図 4 に示すように、候補指数格納部 402、事前計算モジュール 403、事前計算数値格納部 404、分割モジュール 405、逐次処理モジュール 406 及び計算結果格納部 407 を有する。各モジュール 403、405、407 は、CPU 103 が所定のプログラムを実行することで実現される機能単位（モジュール）を表わす。

#### 【0034】

情報処理装置 100 に対しては、外部から入力値  $x$ 、 $N$  (400) 及び  $e$  (401) が入力される。情報処理装置 100 は、入力値からべき乗剰余演算を行い結果として  $c = x^e \pmod{N}$  (408) を出力する装置である。入力値  $N$  が入力されない場合には  $c =$  べき乗演算  $x^e$  を行うが、これはべき乗剰余演算の特殊な場合と考えられるため、第 1 の実施の形態では、べき乗剰余演算  $x^e \pmod{N}$  を行う装置の説明のみを行う。

#### 【0035】

候補指数格納部 402 には、あらかじめ、 $\{(1), (101), (10101), \dots\}$  (2進数表示) のように  $1[01]_L$  (ただし  $[xy]_i$  は  $i$  回  $xy$  を繰り返したもの) の形式を持つ数値が含まれている。事前計算モジュール 403 は、入力値 (400) と候補指数格納部 402 から事前計算を行い、事前計算結果を、例えば HD 106 にある、事前計算数値格納部 404 に格納する。分割モジュール 405 は、入力値 401 を分割し、入力値 401 および分割された数値を、例えば HD 106 に格納する。逐次処理モジュール 406 は、例えば HD 106 にある計算結果格納部 407 を逐次処理して、計算結果 408 を例えば HD 106 に格納する。計算結果はモニタ 102、FD 109、ネットワーク I/F



F114、プリンタ115などを通して出力される。

#### 【0036】

図5は、図4に示した構成を持つ情報処理装置100におけるべき乗剰余演算の処理手順を説明するためのフローチャートである。例えば、CPU103は、図3のフローチャートに対応する処理プログラムを読み出して実行する。これにより、情報処理装置100は次のように動作する。

#### 【0037】

ステップS500:

入力された入力値  $e$  ( $k$  ビット長) を  $\sum_{i=0, \dots, k-1} 2^i * e\_i$  ( $e\_i$  は0または1) のように2進表現しておく。入力値  $x$ ,  $N$ ,  $e$  は例えばHD106に格納される。

#### 【0038】

ステップS501:

入力値  $x$ ,  $N$  から、候補指数格納部402にある  $L$  個の候補指数  $\{l\_i\}$  ( $0 \leq i \leq L-1$ ) のそれぞれに対し、 $x^{l\_i}$  を事前に計算し、事前計算数値格納部404に格納する。

#### 【0039】

ステップS502:

指数  $e$  (ビット長  $k$ ) を候補指数  $\{l\_i\}$  のいずれかと一致するように複数の数値  $\{f\_i\}$  ( $0 \leq i \leq F-1$ ) に分割する。このとき  $f\_i$  のビット長を  $b\_i$  とすると、 $k = \sum_{i=0, \dots, F-1} b\_i$  となるように分割しておく。

#### 【0040】

ステップS503:

まず計算結果格納部407に、 $C := x^{f\_0} \pmod{N}$  とおく。さらに  $f\_i$  ( $0 \leq i \leq F-1$ ) ごとに次のような逐次処理を行う。

for  $i=1$  to  $F-1$

1)  $C := C^2^{b\_i} \pmod{N}$

2) if  $f\_i \neq 0$  then  $C := C * x^{f\_i} \pmod{N}$

N)



## 【0041】

ステップS504:

ステップS503で得られた出力値  $c = x^e \pmod{N}$  を外部に出力する。  
。

## 【0042】

図6、図7および図8は  $e = 1101101110001010001$  の場合の処理例を表す。図6はステップS501における加算鎖の形成方法をあらわしている。 $x \rightarrow x^2 \rightarrow x^4 \rightarrow x^5 \rightarrow x^{10} \rightarrow x^{20} \rightarrow x^{21} \rightarrow x^{42} \rightarrow x^{84} \rightarrow x^{85} \rightarrow \dots$  という処理を経て  $x^5$ 、 $x^{21}$ 、 $x^{85}$  など候補指数  $\{l_i\}$  のそれぞれに対する、 $x^{l_i}$  が計算される。図7はステップS502の処理に対応し、 $e$  を  $f_0 = (1)$ 、 $f_1 = (101)$  などと分割していることを示している。図8はステップS503の処理に対応した計算過程を表している。

## 【0043】

(第2の実施の形態)

上記第1の実施の形態では、候補指数として  $1[01]_L$  の形式の数値のみ扱ったが、本実施形態では  $(11)$  という数値も候補として扱うことによって、第1の実施の形態よりも計算量を減らすことを実現する。

## 【0044】

図9は、第1の実施の形態と同様に  $e = 1101101110001010001$  の場合の処理例であり、ステップS501に相当する事前計算処理における加算鎖の形成方法をあらわしている。図6との違いは、 $x \rightarrow x^2 \rightarrow x^4 \rightarrow x^5$  ではなく  $x \rightarrow x^3 \rightarrow x^5$  と算出している点である。加算鎖が短くなったことに加え、図10のように  $e$  の指数分割においても、分割後数値の個数を少なくすることができ、べき乗剰余演算の計算量を削減することができる。

## 【0045】

(第3の実施の形態)

上記第1及び第2の実施の形態では、 $e$  の指数分割において、分割ビット列が重複するような分割を許していなかった。本実施形態では、分割する際にビット列中の  $(10)$  を  $(01)$  と  $(01)$  の2つに分割することで計算量を削減する



ことを実現する。

#### 【0046】

図11は、第1及び第2の実施の形態と同様に  $e = 1101101110001010001$  の場合の処理例である。 $e$  の先頭3ビット110のうち後半2ビット10を2つの01, 01に分割し、その片方を先頭1ビットに付加することで101を得る。分割されたもう片方の01は、残りのビットに付加される。このような分割処理を繰り返すことにより候補指数の出現確率を高め、ステップS503の逐次処理回数を削減することができる。

#### 【0047】

このときステップS503で使用される  $b\_i$  は  $f\_i$  のビット長をそのまま利用するのではなく、 $f\_i$  どうしの重なりを考慮せねばならない。図11においては、先頭7ビット(1101101)の分割は、 $f\_0 = (101)$ ,  $b\_0 = 2$ ,  $f\_1 = (10101)$ ,  $b\_1 = 1$ ,  $f\_2 = (1)$ ,  $b\_2 = 4$  となり、 $b\_0 + b\_1 + b\_2 = 7$  のようにビット長が一致するように  $b\_i$  を定める必要がある。 $b\_i$  は  $f\_i$  のビット長から次の  $f\_(i+1)$  との重なるビット長を引いた数を利用すればよい。

#### 【0048】

例として指数候補が  $\{(0), (1), (11), (101)\}$  である場合の入力値  $e$  を処理する場合で説明する。指数の値に対する  $f\_i$ ,  $b\_i$  の対と、変数  $sht$  に格納される数値の表を図12に示す。また、 $b\_i$  を求める処理手順のフローチャートを図13に示す。

#### 【0049】

ステップS1301では、入力値  $e$  の先頭3ビットにより処理が分類される。110および111の場合にはさらに1ビット分が読み込まれ、図12に従って処理される。ステップS1302では図13に記載されているとおりに分類ビット列として  $f\_i$  と  $b\_i$  を追加し、前ステップS1301で3ビット読み込んだ場合には3ビット分、4ビット読み込んだ場合には4ビットシフトさせる。ステップS1303では処理ビットが1かどうかで処理を分岐し、0の場合にはステップS1304のように変数  $sht$  を1つ増加させ1ビットシフトする。こ

の処理を先頭ビットが1になるまで繰り返し、ステップS1305を実行する。最終的にすべてのビットを読み込んだかどうかをステップS1306で判断し、読み終えた場合には終了する。分割された  $f\_i$  ,  $b\_i$  の処理はステップS503と同様であるため省略する。

#### 【0050】

##### (第4の実施の形態)

入力された  $e$  によっては事前計算処理が不要の場合が考えられる。例えば  $e = 3$  などビット長が小さい場合、もしくは  $e = 2^{100}$  など2進数表現したときに1が立っているビット数が少ない場合である。この問題に対処するために、入力値  $e$  に対して乗算計算回数を概算することで、事前計算が必要か不要かを選択することができ、ステップS501の処理を無くすることができる。また、 $e$  の分割方法が複数ある場合に、それぞれにおける乗算計算回数を概算して、どの分割方法を採用するかを選択するようにしてもよい。すなわち、概算された乗算計算回数により、指数を分割して計算するか否か、どのように分割するかを制御することができる。

#### 【0051】

また、乗算計算回数を概算する際に、乗算が2乗演算の場合とそうでない場合によって重み付けを行う方法も有効である。High-Speed RSA Implementation, RSA Laboratories, 1994によると、2乗演算は、異なる数値どうしの乗算にくらべ、計算量が少ないことが知られている。例えば、2乗演算の場合には0.8回、そうでない場合には1回とカウントする例が挙げられる。

#### 【0052】

##### (第5の実施の形態)

図14は、図1の情報処理装置100において、特徴とする機能（情報分散処理の機能）に着目して図示したものである。情報処理装置100は、図14に示すように、候補指数格納部402、事前計算モジュール403、事前計算数値格納部404、分割モジュール405、逐次処理モジュール406及び計算結果格納部407を有する。各モジュール403、405、407は、CPU103が



所定のプログラムを実行することで実現される機能単位（モジュール）を表わす。

#### 【0053】

情報処理装置100に対しては、外部から入力値 $x$ 、 $N$ （400）及び $e$ （401）が入力される。情報処理装置100は、入力値からべき乗剰余演算を行い結果として $c = x^e \pmod{N}$ （408）を出力する装置である。入力値 $N$ が入力されない場合には $c =$ べき乗演算 $x^e$ を行うが、これはべき乗剰余演算の特殊な場合と考えられるため、第5の実施の形態では、べき乗剰余演算 $x^e \pmod{N}$ を行う装置の説明のみを行う。

#### 【0054】

候補指数格納部402には、あらかじめ、 $\{(0), (1), (11), (101), (1011), (1101), (10101), (101011), (110101), \dots\}$ （2進数表示）のように $1[01]_L$ または $11[01]_L$ または $1[01]_L1$ （ただし $[xy]_i$ は $i$ 回 $xy$ を繰り返したもの）の形式を持つ数値が含まれている。

#### 【0055】

事前計算モジュール403は、入力値（400）と候補指数格納部402から事前計算を行い、事前計算結果を、例えばHD106にある、事前計算数値格納部404に格納する。分割モジュール405は、入力値401を分割し、入力値401および分割された数値を、例えばHD106に格納する。逐次処理モジュール406は、例えばHD106にある計算結果格納部407を逐次処理して、計算結果408を例えばHD106に格納する。計算結果はモニタ102、FD109、ネットワークI/F114、プリンタ115などを通して出力される。

#### 【0056】

図14に示した構成による情報処理装置100の動作における情報分散処理の手順は、図5に示したフローチャートに従う。例えば、CPU103は、図5のフローチャートに従った処理プログラムを読み出して実行する。これにより、情報処理装置100は次のように動作する。

#### 【0057】

ステップ S500:

入力された入力値  $e$  ( $k$  ビット長) を  $\sum_{i=0, \dots, k-1} 2^i * e_i$  ( $e_i$  は 0 または 1) のように 2 進表現しておく。入力値  $x$ ,  $N$ ,  $e$  は例えば HD 106 に格納される。

【0058】

ステップ S501:

入力値  $x$ ,  $N$  から、候補指数格納部 402 にある  $L$  個の候補指数  $\{l_i\}$  ( $0 \leq i \leq L-1$ ) のそれぞれに対し、 $x^{l_i}$  を事前に計算し、事前計算数値格納部 404 に格納する。

【0059】

事前計算数値格納部 404 の内部には、計算後結果を格納するための  $F_1()$ ,  $F_2()$ ,  $F_3()$ ,  $F_4()$  の 4 つの配列領域 411 ~ 414 (それぞれの配列の長さは  $Q$ ) が用意されている。図 18 は、411 ~ 414 の配列領域に値を格納する処理手順を示すフローチャートである。

【0060】

まず、ステップ S901 により、それぞれの配列領域には  $F_1(0) = x$ ,  $F_2(0) = 1$ ,  $F_3(0) = x$ ,  $F_4(0) = x$  が初期値として設定され、変数  $i$  に 0 が格納される。

【0061】

次にステップ S902 により、 $F_1(i) = F_2(i-1) * F_4(i-1) \pmod{N}$  が格納される。同様にステップ S903 により  $F_2(i) = F_1(i) * F_3(i-1) \pmod{N}$ , ステップ S904 により  $F_3(i) = F_2(i) * F_3(i-1) \pmod{N}$ , ステップ S905 により  $F_4(i) = F_1(i) * F_2(i) \pmod{N}$  が格納され、ステップ S906 により変数  $i$  が  $Q-1$  と一致するかどうか判定する。一致しない場合はステップ S907 で変数  $i$  を 1 つ増加させて、ステップ S902 に戻る。一致する場合は終了である。

【0062】

また、ステップ S904 とステップ S905 は、逐次的に行うのではなく、並

列処理させることが可能であり、並列処理を行うことで高速化が可能である。

### 【0063】

ステップS502:

指数  $e$  (ビット長  $k$ ) を候補指数  $\{l\_i\}$  のいずれかと一致するように複数の数値  $\{f\_i\}$  ( $0 \leq i \leq F-1$ ) に分割する。このとき  $f\_i$  のビット長を  $b\_i$  とすると、 $k = \sum_{i=0, \dots, F-1} b\_i$  となるように分割しておく。

### 【0064】

ステップS503:

まず計算結果格納部407に、 $C := x^{f\_0} \pmod{N}$  とおく。さらに  $f\_i$  ( $0 \leq i \leq F-1$ ) ごとに次のような逐次処理を行う。

for  $i = 1$  to  $F-1$

1)  $C := C^2 \wedge b\_i \pmod{N}$

2) if  $f\_i \neq 0$  then  $C := C * x^{f\_i} \pmod{N}$

### 【0065】

ステップS504:

ステップS503で得られた出力値  $c = x^e \pmod{N}$  を外部に出力する。

### 【0066】

図15は、本実施形態における加算鎖の構成方法を表している。前述のように、候補指数は2進数表示したとき  $1[01]_L$  または  $11[01]_L$  または  $1[01]_L 1$  (ただし  $[xy]_i$  は  $i$  回  $xy$  を繰り返したものの) の形式を持つが、これらの候補指数を効率的に算出する方法を以下に示す。

### 【0067】

$f_1()$ ,  $f_2()$ ,  $f_3()$ ,  $f_4()$  の4つの関数は  $f_1(0) = 1$ ,  $f_2(0) = 0$ ,  $f_3(0) = 1$ ,  $f_4(0) = 1$  を初期値として  $f_1(i) = f_2(i-1) + f_4(i-1)$ ,  $f_2(i) = f_1(i) + f_3(i-1)$ ,  $f_3(i) = f_2(i) + f_3(i-1)$ ,  $f_4(i) = f_1(i) + f_2(i)$

を満たすように循環的に計算を行っていく。計算順序は、 $f_1(1) \rightarrow f_2(1) \rightarrow f_3(1) \rightarrow f_4(1) \rightarrow f_1(2) \rightarrow f_2(2) \rightarrow f_3(2) \rightarrow f_4(2) \rightarrow \dots$  のように行う。このとき、 $f_1(i) = 1[01]_i$ ,  $f_2(i) = 10[00]_i$ ,  $f_3(i) = 11[01]_i$ ,  $f_4(i) = 1[01]_i 1$  という形式となる。このようにして、 $\{1, 2, 3, 5, 8, 11, 13, 21, 32, 43, 53, 85, 128, 171, 213, 314, \dots\}$  という加算鎖を構成することができる。

### 【0068】

図16および図17は、候補指数の最大ビット長  $W=4$  (つまり指数候補は  $\{(1), (11), (101), (1011), (1101)\}$  である),  $e = 1101101110001010001$  の場合の処理例を表す。まずステップ S501 に対応して、候補指数  $\{1\_i\}$  のそれぞれに対する、 $x^{1\_i}$  が計算される。図7はステップ S502 の処理に対応し、 $e$  を  $f\_0 = (1101)$ ,  $f\_1(1011)$ ,  $f\_2 = (11)$  などと分割していることを示している。図8はステップ S503 の処理に対応した計算過程を表している。

### 【0069】

#### (第6の実施の形態)

上記第5の実施の形態では、 $e$  の指数分割において、分割ビット列が重複するような分割を許していなかったが、第3の実施の形態と同様に、分割する際にビット列中の  $(10)$  を  $(01)$  と  $(01)$  の2つに分割することで計算量を削減することが実現できる。図19は、図12の表及び図13記載のフローチャートを用いて、 $e = 111110111000110100111$  を分割した例である。

### 【0070】

#### (その他の実施の形態)

本発明は、複数の機器 (例えばホストコンピュータ等) から構成されるシステムの一部として適用しても、一つの機器からなるものの一部に適用してもよい。

### 【0071】

また、上述した実施の形態の機能を実現するべく各種のデバイスを動作させる

ように、該各種デバイスと接続された装置或いはシステム内のコンピュータに対し、上記実施の形態の機能を実現するためのソフトウェアのプログラムコードを供給し、そのシステム或いは装置のコンピュータ（CPU或いはMPU）に格納されたプログラムに従って上記各種デバイスを動作させることによって実施したものも、本発明の範疇に含まれる。

#### 【0072】

また、この場合、上記ソフトウェアのプログラムコード自体が上述した実施の形態の機能を実現することになり、そのプログラムコード自体は本発明を構成する。そのプログラムコードの伝送媒体としては、プログラム情報を搬送波として伝搬させて供給するためのコンピュータネットワーク（LAN、インターネット等のWAN、無線通信ネットワーク等）システムにおける通信媒体（光ファイバ等の有線回線や無線回線等）を用いることができる。

#### 【0073】

さらに、上記プログラムコードをコンピュータに供給するための手段、例えばかかるプログラムコードを格納した記録媒体は本発明を構成する。かかるプログラムコードを記憶する記録媒体としては、例えばフレキシブルディスク、ハードディスク、光ディスク、光磁気ディスク、CD-ROM、磁気テープ、不揮発性のメモリカード、ROM等を用いることができる。

#### 【0074】

また、コンピュータが供給されたプログラムコードを実行することにより、上述の実施の形態の機能が実現されるだけでなく、そのプログラムコードがコンピュータにおいて稼働しているOS（オペレーティングシステム）或いは他のアプリケーションソフト等と共同して上述の実施の形態の機能が実現される場合にもかかるプログラムコードは本発明の実施の形態に含まれることはいうまでもない。

#### 【0075】

さらに、供給されたプログラムコードがコンピュータの機能拡張ボードやコンピュータに接続された機能拡張ユニットに備わるメモリに格納された後、そのプログラムコードの指示に基づいてその機能拡張ボードや機能拡張ユニットに備わ

るCPU等が実際の処理の一部又は全部を行い、その処理によって上述した実施の形態の機能が実現される場合にも本発明に含まれることはいうまでもない。

#### 【0076】

以下、上記実施形態に係わる本発明の特徴を整理する。

#### 【0077】

特徴1.

2つの整数 $x$ および $e$ から $x^e$ を計算すべき乗演算装置において、

2つの整数 $x$ および $e$ を入力する入力手段と、

$L$ 個の候補指数  $\{l\_i\}$  ( $0 \leq i \leq L-1$ ) を格納する候補指数格納手段と

、  
前記候補指数格納手段に格納されている候補指数  $\{l\_i\}$  のそれぞれに対し、  
入力された整数 $x$ から $x^{l\_i}$ を事前に計算しておく事前計算手段と、

事前計算された数値 $x^{l\_i}$ を格納する事前計算数値格納手段と、

入力された整数 $e$ を前記候補指数  $\{l\_i\}$  のいずれかと一致するように複数の  
数値  $\{f\_i\}$  ( $0 \leq i \leq F-1$ ) に分割する分割手段と、

計算結果 $c$ を格納する計算結果格納手段と、

分割された数値  $\{f\_i\}$  ( $0 \leq i \leq F-1$ ) について、事前計算された数値  
 $x^{l\_i}$ を用いて計算結果 $c$ を逐次更新する逐次処理手段と、

全ての前記数値  $\{f\_i\}$  について更新後の計算結果 $c$ を $x^e$ として出力する  
出力手段と

を有することを特徴とするべき乗演算装置。

#### 【0078】

特徴2.

3つの整数 $x$ 、 $e$ および $N$ から $x^e \pmod{N}$ を計算すべき乗演算装置  
において、

3つの整数 $x$ 、 $e$ および $N$ を入力する入力手段と、

$L$ 個の候補指数  $\{l\_i\}$  ( $0 \leq i \leq L-1$ ) を格納する候補指数格納手段と

、  
前記候補指数格納手段に格納されている候補指数  $\{l\_i\}$  のそれぞれに対し

、入力された整数  $x$  から  $x^{\wedge} \{1\_i\}$  を事前に計算しておく事前計算手段と、  
 事前計算された数値  $x^{\wedge} \{1\_i\}$  を格納する事前計算数値格納手段と、  
 入力された整数  $e$  を前記候補指数  $\{1\_i\}$  のいずれかと一致するように複数の  
 数値  $\{f\_i\}$  ( $0 \leq i \leq F-1$ ) に分割する分割手段と、  
 計算結果  $c$  を格納する計算結果格納手段と、  
 分割された数値  $\{f\_i\}$  ( $0 \leq i \leq F-1$ ) について、事前計算された数値  
 $x^{\wedge} \{1\_i\}$  を用いて計算結果  $c$  を逐次更新する逐次処理手段と、  
 全ての前記数値  $\{f\_i\}$  について更新後の計算結果  $c$  を  $x^e \pmod{N}$   
 ) として出力する出力手段と  
 を有することを特徴とするべき乗演算装置。

### 【0079】

特徴3.

前記逐次処理手段は、

前記計算結果格納手段内に、前記計算結果  $c$  の初期値として  $f\_0$  を設定する  
 初期化手段と、

前記分割数値  $\{f\_i\}$  ( $1 \leq i \leq F-1$ ) のそれぞれにつき、 $f\_i$  を2進  
 数表示したときのビット長分  $c := c^2$  の更新処理を行い、さらに  $c := c * f$   
 $\_I$  の更新処理を行う更新処理手段と

を有することを特徴とする特徴1または2に記載の情報処理装置。

### 【0080】

特徴4.

前記候補指数格納手段に格納される候補指数は、2進数表示が(0)または1  
 $[0\ 1]_L$  (ただし  $[x\ y]_i$  は  $i$  回  $x\ y$  を繰り返したもの) の形式であること  
 を特徴とする特徴1または2に記載の情報処理装置。

### 【0081】

特徴5.

前記候補指数格納手段に格納された候補指数は、2進数表示が、(0)または  
 $(1\ 1)$  または  $1\ [0\ 1]_L$  (ただし  $[x\ y]_i$  は  $i$  回  $x\ y$  を繰り返したもの)  
 の形式であることを特徴とする特徴1または2に記載の情報処理装置。

## 【0082】

特徴6.

前記分割手段は、分割後数値を2進数表示したとき、ビット列中の(10)を(01)と(01)の2つに分割して、 $\{f\_i\}$ の重複を許す分割方法であり、

前記逐次処理手段における $c := c^2$ の更新処理を、 $f\_i$ のビット長分行うのではなく、重複しているビット列の長さ分は更新処理を行わないことを特徴とする特徴1または2に記載の情報処理装置。

## 【0083】

特徴7.

前記分割手段による分割に応じた乗算回数を概算する乗算回数概算手段と、当該概算された乗算回数に基づいて、前記分割手段による分割を制御する分割制御手段とを有することを特徴とする特徴1または2に記載の情報処理装置。

## 【0084】

特徴8.

前記乗算回数概算手段は、異なる2値の乗算と同一値同士の乗算とで、異なる重みをつけて乗算回数を概算することを特徴とする特徴7に記載の情報処理装置。

## 【0085】

特徴9.

前記候補指数格納手段に格納される候補指数の個数 $L$ を、入力値 $e$ のビット長により増減させることを特徴とする特徴1または2に記載の情報処理装置。

## 【0086】

特徴10.

前記候補指数格納手段に格納された候補指数は、0であるかまたは、2進数表示で $W$ ビット数以下であって、 $1[01]_L$  (ただし $[xy]_i$ は $xy$ を $i$ 回繰り返したものの)の形式であるか、 $11[01]_L$ の形式であるか、または $1[01]_L 1$ の形式であることを特徴とする特徴1または2に記載の情報処理装置。

## 【0087】



特徴 11.

前記事前計算手段は、

候補指数を表現する  $f_1()$ ,  $f_2()$ ,  $f_3()$ ,  $f_4()$  の 4 つの関数を利用し、

$f_1(0) = 1$ ,  $f_2(0) = 0$ ,  $f_3(0) = 1$ ,  $f_4(0) = 1$  を初期値として

$f_1(i) = f_2(i-1) + f_4(i-1)$ ,  $f_2(i) = f_1(i) + f_3(i-1)$ ,  $f_3(i) = f_2(i) + f_3(i-1)$ ,  $f_4(i) = f_1(i) + f_2(i)$  を満たすように循環的に計算を行うことで、 $f_1(i) = 1 [01]_i$ ,  $f_2(i) = 10 [00]_i$ ,  $f_3(i) = 11 [01]_i$ ,  $f_4(i) = 1 [01]_i$  という形式を得ることで加算鎖を構成し、

$x f_2(i-1)$  と  $x f_4(i-1)$  の積から  $x f_1(i)$  を、 $x f_1(i)$  と  $x f_3(i-1)$  の積から  $x f_2(i)$  を、 $x f_2(i)$  と  $x f_3(i-1)$  の積から  $x f_3(i)$  を、 $x f_1(i)$  と  $x f_2(i)$  の積から  $x f_4(i)$  を計算し、

当該計算結果を前記計算結果格納手段に格納することを特徴とする特徴 10 に記載の情報処理装置。

【0088】

特徴 12.

前記計算結果格納手段は、

計算後結果を格納する  $F_1()$ ,  $F_2()$ ,  $F_3()$ ,  $F_4()$  の 4 つの配列領域を持ち、

$F_1(0) = x$ ,  $F_2(0) = 1$ ,  $F_3(0) = x$ ,  $F_4(0) = x$  を初期値として設定し、

前記事前計算手段は、

$F_1(i) = F_2(i-1) * F_4(i-1)$ ,  $F_2(i) = F_1(i) * F_3(i-1)$ ,  $F_3(i) = F_2(i) * F_3(i-1)$ ,  $F_4(i) = F_1(i) * F_2(i)$  を満たすように循環的に計算し、

当該計算結果を前記計算結果格納手段に格納することを特徴とする特徴 10 に

記載の情報処理装置。

【0089】

特徴13.

前記候補指数格納手段に格納される候補指数は、前記整数  $e$  のビット数に応じて  $W$  を変化させることを特徴とする特徴10に記載の情報処理装置。

【0090】

特徴14.

2つの整数  $x$  および  $e$  から  $x^e$  を計算すべき乗演算方法において、

2つの整数  $x$  および  $e$  を入力する入力工程と、

候補指数格納部に格納されている  $L$  個の候補指数  $\{l\_i\}$  ( $0 \leq i \leq L-1$ ) のそれぞれに対し、入力された整数  $x$  から  $x^{l\_i}$  を事前に計算して事前計算数値格納部に格納しておく事前計算工程と、

入力された整数  $e$  を前記候補指数  $\{l\_i\}$  のいずれかと一致するように複数の数値  $\{f\_i\}$  ( $0 \leq i \leq F-1$ ) に分割する分割工程と、

分割された数値  $\{f\_i\}$  ( $0 \leq i \leq F-1$ ) について、事前計算された数値  $x^{l\_i}$  を用いて、計算結果格納部に格納されている計算結果  $c$  を逐次更新する逐次処理工程と、

全ての前記数値  $\{f\_i\}$  について更新後の計算結果  $c$  を  $x^e$  として出力する出力工程と

を有することを特徴とするべき乗演算方法。

【0091】

特徴15.

3つの整数  $x$ 、 $e$  および  $N$  から  $x^e \pmod{N}$  を計算すべき乗演算方法において、

3つの整数  $x$ 、 $e$  および  $N$  を入力する入力工程と、

候補指数格納部に格納されている  $L$  個の候補指数  $\{l\_i\}$  ( $0 \leq i \leq L-1$ ) のそれぞれに対し、入力された整数  $x$  から  $x^{l\_i}$  を事前に計算して事前計算数値格納部に格納しておく事前計算工程と、

入力された整数  $e$  を前記候補指数  $\{l\_i\}$  のいずれかと一致するように複数

の数値  $\{f\_i\}$  ( $0 \leq i \leq F-1$ ) に分割する分割工程と、

分割された数値  $\{f\_i\}$  ( $0 \leq i \leq F-1$ ) について、事前計算された数値  $x^{\wedge} \{l\_i\}$  を用いて、計算結果格納部に格納されている計算結果  $c$  を逐次更新する逐次処理工程と、

全ての前記数値  $\{f\_i\}$  について更新後の計算結果  $c$  を  $x^e \pmod{N}$  として出力する出力工程と

を有することを特徴とするべき乗演算方法。

#### 【0092】

特徴 16.

2つの整数  $x$  および  $e$  から  $x^e$  を計算するべき乗演算をコンピュータに実行させるためのコンピュータ読み取り可能なプログラムにおいて、

2つの整数  $x$  および  $e$  を入力する入力工程と、

候補指数格納部に格納されている  $L$  個の候補指数  $\{l\_i\}$  ( $0 \leq i \leq L-1$ ) のそれぞれに対し、入力された整数  $x$  から  $x^{\wedge} \{l\_i\}$  を事前に計算して事前計算数値格納部に格納しておく事前計算工程と、

入力された整数  $e$  を前記候補指数  $\{l\_i\}$  のいずれかと一致するように複数の数値  $\{f\_i\}$  ( $0 \leq i \leq F-1$ ) に分割する分割工程と、

分割された数値  $\{f\_i\}$  ( $0 \leq i \leq F-1$ ) について、事前計算された数値  $x^{\wedge} \{l\_i\}$  を用いて、計算結果格納部に格納されている計算結果  $c$  を逐次更新する逐次処理工程と、

全ての前記数値  $\{f\_i\}$  について更新後の計算結果  $c$  を  $x^e$  として出力する出力工程と

を有することを特徴とするプログラム。

#### 【0093】

特徴 17.

3つの整数  $x$ 、 $e$  および  $N$  から  $x^e \pmod{N}$  を計算するべき乗演算をコンピュータに実行させるためのコンピュータ読み取り可能なプログラムにおいて、

3つの整数  $x$ 、 $e$  および  $N$  を入力する入力工程と、

候補指数格納部に格納されている  $L$  個の候補指数  $\{l\_i\}$  ( $0 \leq i \leq L-1$ ) のそれぞれに対し、入力された整数  $x$  から  $x^{l\_i}$  を事前に計算して事前計算数値格納部に格納しておく事前計算工程と、

入力された整数  $e$  を前記候補指数  $\{l\_i\}$  のいずれかと一致するように複数の数値  $\{f\_i\}$  ( $0 \leq i \leq F-1$ ) に分割する分割工程と、

分割された数値  $\{f\_i\}$  ( $0 \leq i \leq F-1$ ) について、事前計算された数値  $x^{l\_i}$  を用いて、計算結果格納部に格納されている計算結果  $c$  を逐次更新する逐次処理工程と、

全ての前記数値  $\{f\_i\}$  について更新後の計算結果  $c$  を  $x^e \pmod{N}$  ) として出力する出力工程と  
を有することを特徴とするプログラム。

#### 【0094】

##### 【発明の効果】

以上説明したように本発明によれば、 $e$  を2進数表示した際のビット列内にある特徴を持つビット列が登場することを想定して、それらのビット列を候補指数として、それらに関してのみ事前計算を行うことで、事前計算処理量を削減し、従来方式に比べて少ない演算回数で計算することができるべき乗計算方式を提供することができる。

#### 【0095】

また事前計算しておく数値の数が少なくなるので、事前計算数値を記憶するためのテーブルサイズをおさえるメリットもあわせ持ち、テーブル参照のためのメモリ領域を削減することができる。

##### 【図面の簡単な説明】

##### 【図1】

実施形態の情報処理装置の構成を示すブロック図である。

##### 【図2】

従来例である Binary Methodでの処理を表す図である。

##### 【図3】

従来例である Quaternary Methodでの処理を表す図である。

**【図 4】**

第 1 の実施の形態における情報処理装置の機能構成を示すブロック図である。

**【図 5】**

第 1 の実施の形態におけるべき乗剰余演算の処理手順を説明するためのフローチャートである。

**【図 6】**

第 1 の実施の形態における加算鎖の形成方法を表す図である。

**【図 7】**

第 1 の実施の形態における指数分割の例を示す図である。

**【図 8】**

第 1 の実施の形態における逐次計算の例を示す図である。

**【図 9】**

第 2 の実施の形態における加算鎖の形成方法を表す図である。

**【図 1 0】**

第 2 の実施の形態における指数分割の例を示す図である。

**【図 1 1】**

第 3 の実施の形態における指数分割の例を示す図である。

**【図 1 2】**

指数の値に対する  $f\_i$  ,  $b\_i$  の対と、変数  $s h t$  に格納される数値の表を表す図である。

**【図 1 3】**

$b\_i$  を求める処理手順を示すフローチャートである。

**【図 1 4】**

第 5 の実施の形態における情報処理装置の機能構成を示すブロック図である。

**【図 1 5】**

第 5 の実施の形態における加算鎖の形成方法を表す図である。

**【図 1 6】**

第 5 の実施の形態における指数分割の例を示す図である。

**【図 1 7】**

第5の実施の形態における逐次計算の例を示す図である。

【図18】

配列領域に値を格納する処理手順を示すフローチャートである。

【図19】

第6の実施の形態における指数分割の例を示す図である。

【符号の説明】

- 100 情報処理装置
- 102 モニタ
- 103 CPU
- 104 ROM
- 105 RAM
- 106 HD
- 107 ネットワーク接続部
- 108 CDドライブ
- 109 FDドライブ
- 110 DVDドライブ
- 111 インターフェース
- 112 マウス
- 113 キーボード
- 114 インターフェース
- 115 プリンタ
- 116 バス
- 117 インターフェース
- 118 モデム
- 119 インターフェース
- 400 入力値  $x$ 、 $N$
- 401 入力値  $e$
- 402 候補指数格納部
- 403 事前計算モジュール

4 0 4 事前計算数値格納部

4 0 5 分割モジュール

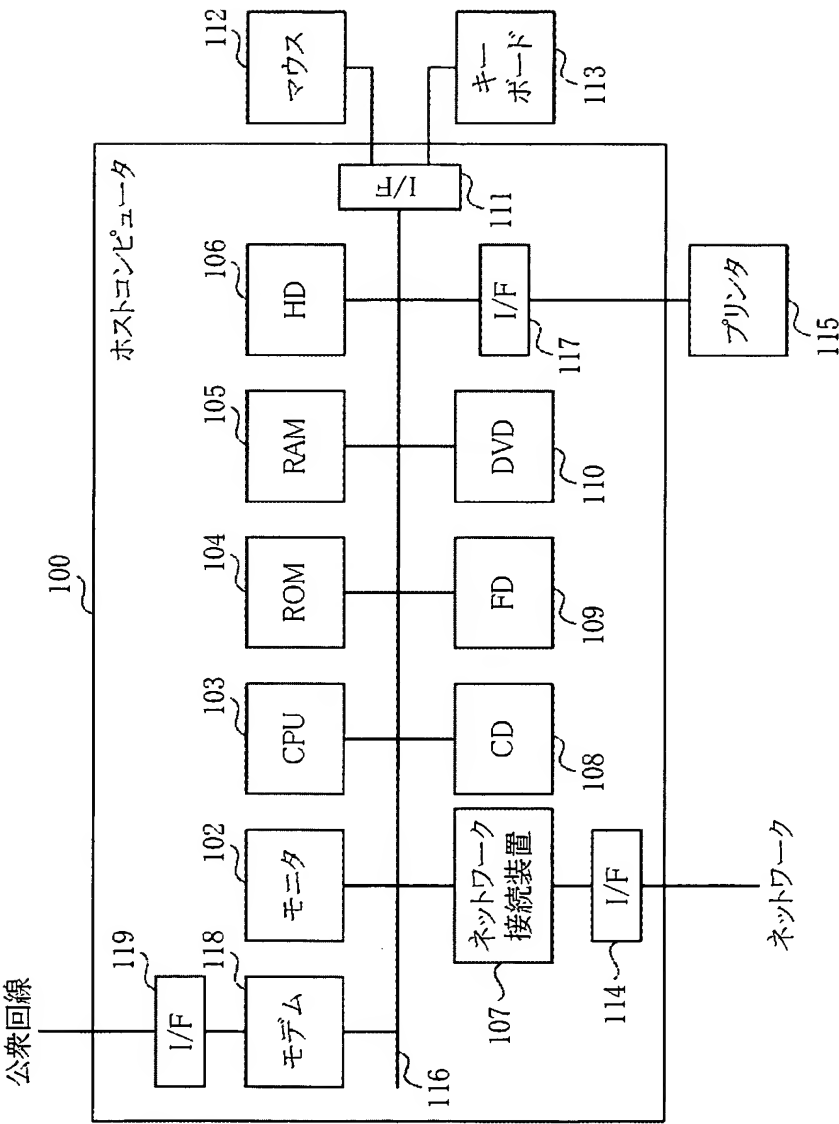
4 0 6 逐次処理モジュール

4 0 7 計算結果格納部

4 0 8 計算結果 c

4 1 1 ~ 4 1 4 配列領域

【書類名】 図面  
【図 1】





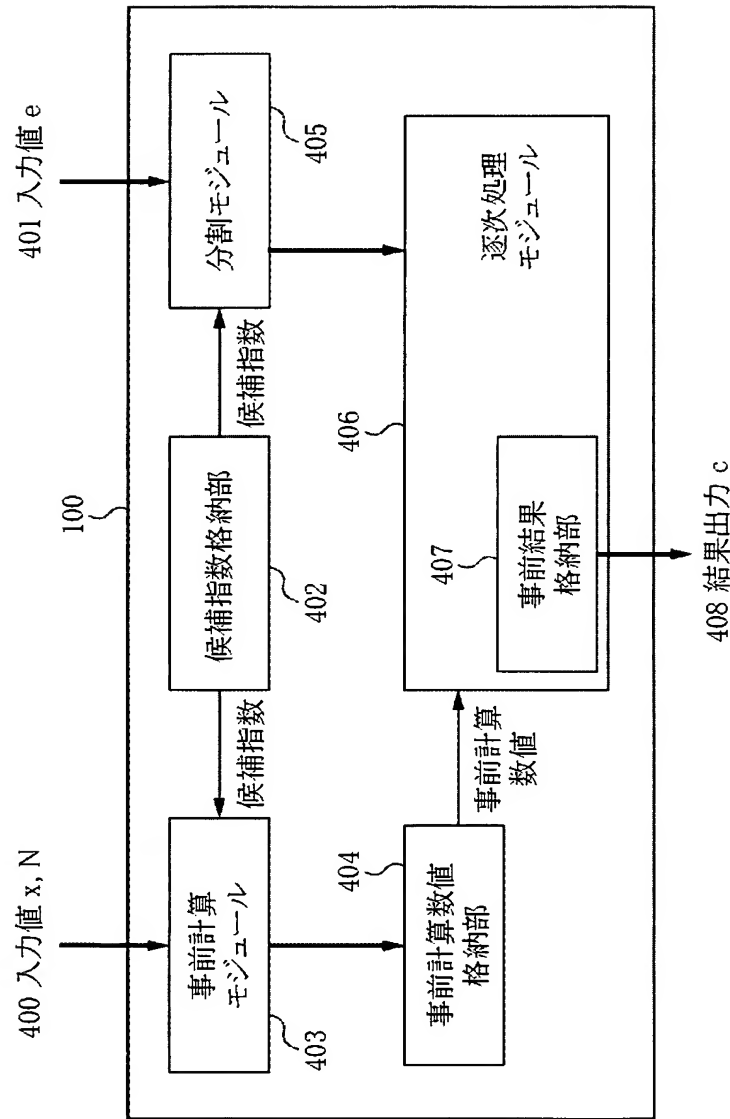
【図 2】

	処理2-1)	処理2-2)
1		$x$
1	$(x)^2=x^2$	$x^2 \cdot x=x^3$
0	$(x^3)^2=x^6$	$x^6$
1	$(x^6)^2=x^{12}$	$x^{12} \cdot x=x^{13}$
1	$(x^{13})^2=x^{26}$	$x^{26} \cdot x=x^{27}$
1	$(x^{27})^2=x^{54}$	$x^{54} \cdot x=x^{55}$

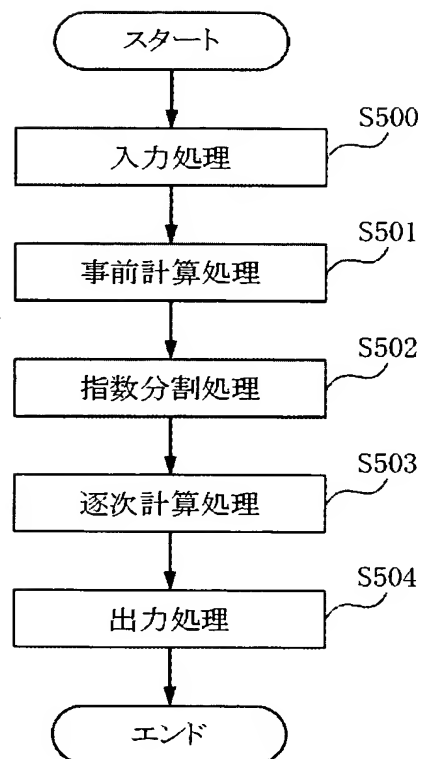
【図 3】

	処理2-1)	処理2-2)
11		$x^3$
01	$(x^3)^4 = x^{12}$	$x^{12} \cdot x = x^{13}$
11	$(x^{13})^4 = x^{52}$	$x^{52} \cdot x^3 = x^{55}$

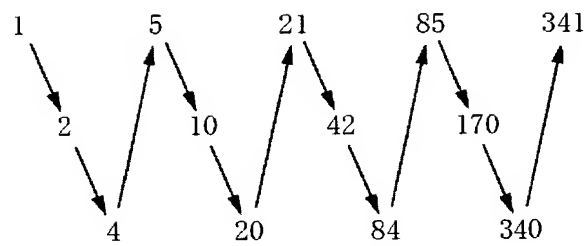
【図 4】



【図 5】



【図 6】



【図 7】

e=1101101110001010001

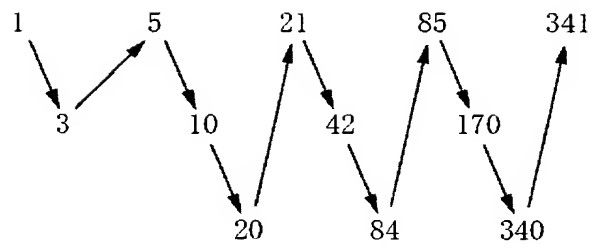
1 101 101 11000 1010001

f<sub>0</sub>      f<sub>1</sub>...

【図 8】

	ステップS503処理1)	処理2)
1		$x^1$
101	$(x^1)^8 = x^8$	$x^8 \cdot x^5 = x^{13}$
101	$(x^{13})^8 = x^{104}$	$x^{104} \cdot x^5 = x^{109}$
1	$(x^{109})^2 = x^{218}$	$x^{218} \cdot x^1 = x^{219}$
1	$(x^{219})^2 = x^{438}$	$x^{438} \cdot x^1 = x^{439}$
0	$(x^{439})^2 = x^{878}$	$= x^{878}$

【図 9】



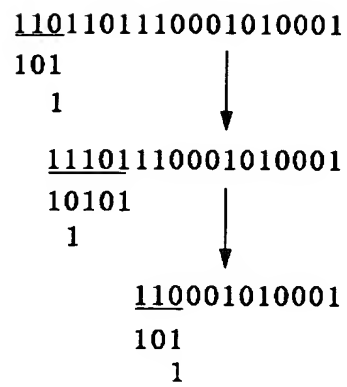


【図 1 0】

e=1101101110001010001

1 101 101 11 000 101 000 1

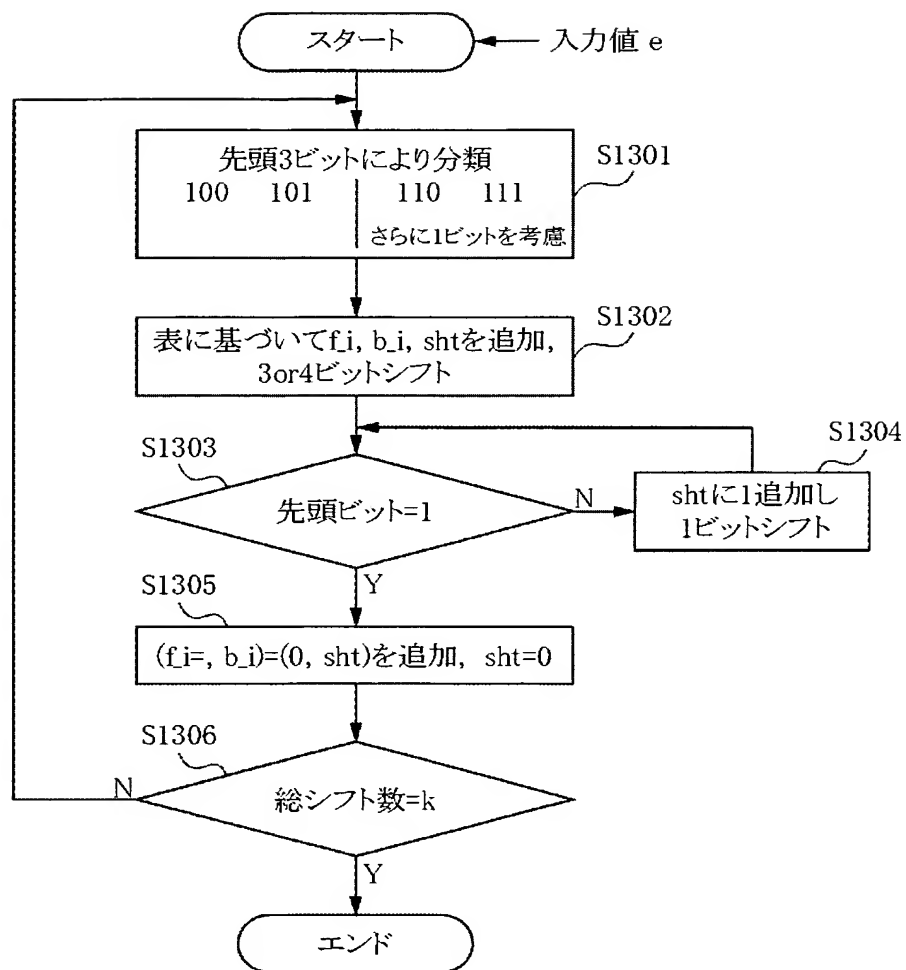
【図 1 1】



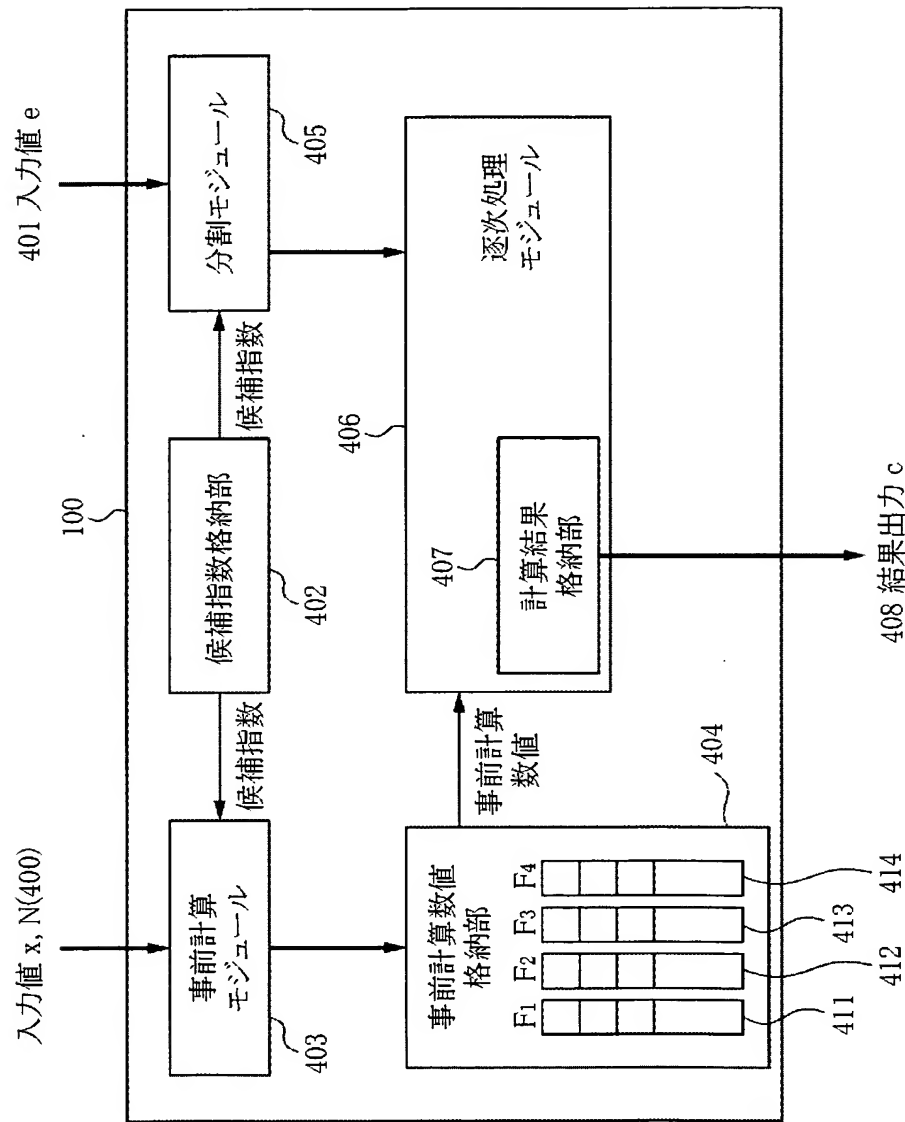
【図 1 2】

		$f_i, b_i$	$f_{(i+1)}, b_{(i+1)}$	sft
100		1, 1	—	2
101		101, 3	—	0
110	0	11, 2	—	2
	1	1, 1	101, 3	0
111	0	101, 1	1, 1	2
	1	101, 1	101, 3	0

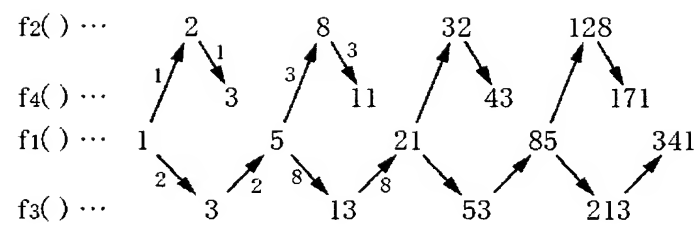
【図 13】



【図 14】



【図 1 5】



【図 1 6】

$e = 1101101111001010001$

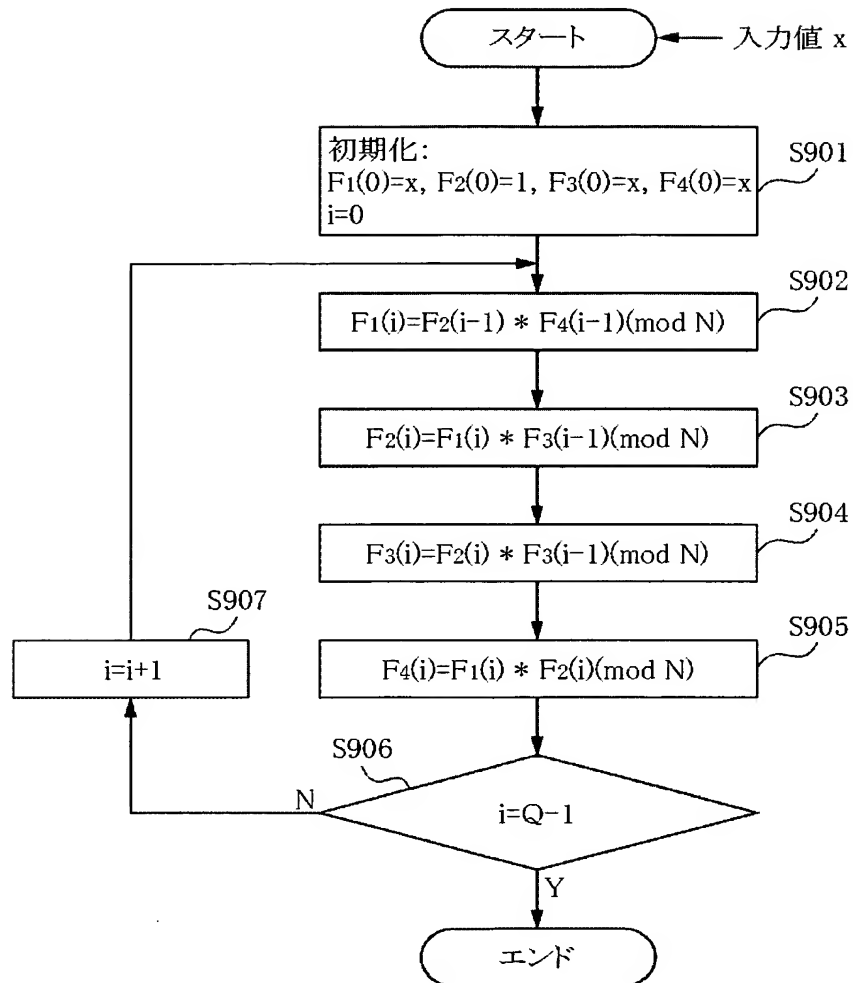
1101 1011 1100 1010001  
 $f_0$   $f_1 \dots$

【図 1 7】

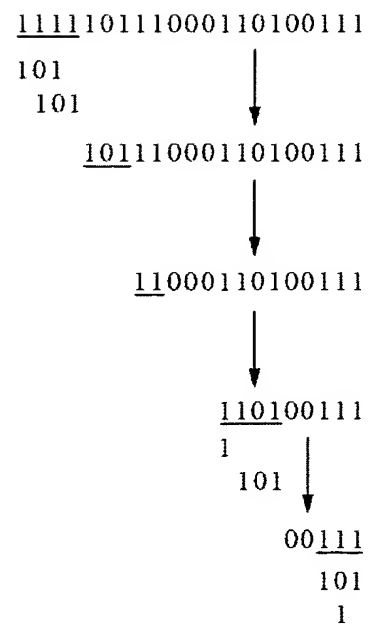
	ステップS503処理1)	処理2)
1101		$x^{13}$
1011	$(x^{13})^{16}=x^{208}$	$x^{208} \cdot x^{11}=x^{219}$
11	$(x^{219})^4=x^{876}$	$x^{876} \cdot x^3=x^{879}$
0	$(x^{879})^2=x^{1758}$	$=x^{1758}$
0	$(x^{1758})^2=x^{3516}$	$=x^{3516}$
101	$(x^{3516})^8=x^{28128}$	$x^{28128} \cdot x^5=x^{28133}$
...	...	...



【図 18】



【図 1 9】



【書類名】 要約書

【要約】

【課題】 事前計算処理量とテーブルサイズを低減させ、少ない演算回数で計算できるべき乗演算装置を提供する。

【解決手段】 入力された2つの整数  $x$  および  $e$  から  $x^e$  を計算するべき乗演算装置において、候補指数格納部 402 に格納された  $L$  個の候補指数  $\{l\_i\}$  ( $0 \leq i \leq L-1$ ) のそれぞれに対し、事前計算モジュール 403 により  $x^{l\_i}$  を事前に計算して事前計算数値格納部 404 に格納しておき、分割モジュール 405 により  $e$  を候補指数  $\{l\_i\}$  のいずれかと一致するように複数の数値  $\{f\_i\}$  ( $0 \leq i \leq F-1$ ) に分割し、逐次処理モジュール 406 により、 $\{f\_i\}$  について、計算結果格納部 407 に格納された計算結果  $c$  を  $x^{l\_i}$  を用いて逐次更新し、全ての  $\{f\_i\}$  について更新後の計算結果  $c$  を  $x^e$  として出力する。

【選択図】 図 4

特願 2 0 0 2 - 3 1 4 5 9 5

出 願 人 履 歴 情 報

識別番号

[ 0 0 0 0 0 1 0 0 7 ]

1. 変更年月日

1 9 9 0 年 8 月 3 0 日

[変更理由]

新規登録

住 所

東京都大田区下丸子3丁目30番2号

氏 名

キャノン株式会社